LELIO: An Auto-Adaptative System to Acquire Domain Lexical Knowledge in Technical Texts

Patrick Saint-Dizier
IRIT-CNRS,
118 route de Narbonne, 31062 Toulouse cedex France,
stdizier@irit.fr

Abstract

In this paper, we investigate some language acquisition facets of an auto-adaptative system that can automatically acquire most of the relevant lexical knowledge and authoring practices for an application in a given domain. This is the LELIO project: producing customized LELIE solutions. Our goal, within the framework of LELIE (a system that tags language uses that do not follow the Constrained Natural Language principles), is to automate the long, costly and error prone lexical customization of LELIE to a given application domain. Technical texts being relatively restricted in terms of syntax and lexicon, results obtained show that this approach is feasible and relatively reliable. By auto-adaptative, we mean that the system learns from a sample of the application corpus the various lexical terms and uses crucial for LELIE to work properly (e.g. verb uses, fuzzy terms, business terms, stylistic patterns). A technical writer validation method is developed at each step of the acquisition.

Keywords: Lexical Acquisition, Constrained Natural Language, Technical Text Authoring

1. Aims and Challenges

Technical documents form a linguistic genre with specific linguistic constraints in terms of lexical realization, syntax, typography and overall document organization, including business or domain dependent aspects. Technical documents cover a large variety of types of documents: procedures, equipment and product manuals, various notices such as security notices, regulations of various types (security, management), requirements and specifications (Hull et ali. 2011). These documents are designed to be easy to read and as efficient and unambiguous as possible for their users. They must leave little space for personal interpretations. For that purpose, they tend to follow relatively strict controlled natural language (CNL hereafter) principles concerning both their form and contents (Fuchs 2012), (Kuhn 2014) (Aurora et ali. 2013). These principles are described in documents called authoring guidelines. These are general purpose statements, norms in e.g. aeronautics, chemistry, or considerations proper to a company. Guidelines contain some details about e.g. the terms which can be used (for example, a limited subset of verbs) or not and syntactic structures which are allowed or preferred. Stylistic considerations may also be introduced, for example concerning the order of complements or subordinate clauses (e.g. condition - main clause - purpose).

These considerations have motivated the development of the LELIE project (Barcellini et al. 2012), (Saint-Dizier 2014), which is a system that detects several types of errors in technical documents, whatever their authoring and revision stages are. Lelie produces **alerts** related to these errors on terms, expressions or constructions that need various forms of improvements, at all levels: lexical, syntactic, semantics and discourse. LELIE deals with errors which are not detected by usual text editors such as MS Word. These errors may indeed be perfectly acceptable expressions in non-technical language.

For each application domain, and for each company, LELIE must be customized. The major part of the customization is the development of the lexical resources and their features, while the error detection rules remain relatively stable over domains. Acquiring and structuring the lexical knowledge of an application domain is a very much costly and time consuming activity, and an error prone process. This is a major limitation to an accurate and efficient customization of applications. The lexicon proper to an application is about 60% of the total lexicon.

In this paper, we investigate, within the framework of LELIE, the different facets of an auto-adaptative system that can acquire most of the relevant lexical knowledge for an application in a given domain. This is the **LELIO** project: producing customized LELIE solutions to an application domain or in accordance with a company authoring practices. Technical texts being relatively restricted in terms of syntax and lexicon, results obtained show that this approach is feasible and relatively reliable, with some form of human validation. By auto-adaptative, we mean that the system:

(1) learns from a sample of the application corpus the various lexical terms and uses crucial in LELIE (e.g. verb uses, fuzzy terms, business terms) and

(2) automatically generates the application oriented lexicon, which is paired with the domain independent lexicon, defined in the kernel of LELIE.

This topic is very vast and has been investigated at the level of the re-usability of resources. This notion of auto-adaptative system has been developed in dialogs, machine translation (Scherrer 2007) and to a lesser extend in lexical tuning in opinion analysis. To the best of our knowledge, nothing has been done in the type of context we consider. Foundational ergonomics principles are investigated in (Ganier et al. 2007), and in (Weiss 2000 and (White et ali. 2009) from a more linguistic and pedagogical perspective. In this paper, we address this variable part of the resources which must be acquired. We concentrate in this paper on two main cases: the acquisition and the character-

ization of fuzzy terms in an application and the acquisition of business terms. We develop in section 5 a style facet of technical authoring by an induction mechanism of the main stylistic practices which can then be used as a guideline or a norm. The techniques developed in this work are based on lexical and syntactic considerations. These techniques are simple but seem to produce relatively satisfactory results.

2. The LELIE Prototype

The LELIE prototype (Version V2.1, code freely available CC BY NC) detects errors related to the non-observation of a recommendation in the authoring guidelines that is considered in various types of technical texts: procedures, specifications or requirements (Hull et ali.,. 2011), (Grady, 2006). For example, according to most CNL principles and guidelines, the use of passives, modals or negation must be avoided in instructions (Kuhn 2014). Specific errors have also been observed for non-native speakers of the language the use (Garnier 2011). LELIE also allows to specify business constraints such as controls on style and the use of business terms. The errors detected by LELIE are typical errors of technical texts, they are not errors in ordinary language.

Error detection in LELIE depends both on the textual genre and on the precise discourse structure that is observed: for example modals are the norm in requirements (Grady 2006) but not in instructions. Titles may contain deverbals which are not frequently admitted in instructions or warnings (Grady 2006). LELIE is parameterized and offers several levels of alerts depending on the a priori error severity level. LELIE and the experiments reported below have been developed on the logic-based <TextCoop> platform (Saint-Dizier 2012, 2014). The output of LELIE is the original text with annotations. Table 1 below shows the major errors found by LELIE, statistics have been realized on 40 pages (about 20 000 words) of proofread technical documents from companies A, B and C:

Error type	Average nb. of errors for 40 pages	A	B	C
fuzzy lexical items	66	44	89	49
deverbals	29	24	14	42
modals in instructions	5	0	12	1
light verb constructions	2	2	2	3
pronouns with unclear reference	22	4	48	2
negation	52	8	109	9
complex discourse structures	43	12	65	50
complex coordinations	19	30	10	17
heavy N+N or noun complements	46	58	62	15
passives	34	16	72	4
future tense	2	2	4	1
sentences too complex	108	16	221	24
irregular enumerative construction rate	average	low	high	average
incorrect references to sections or figures	13	33	22	2

Table 1. Errors found in technical texts for companies A, B and C

These results show that there is on average about one error every 4 lines of text, not counting errors related to business term. This error rate is very high. The alerts produced by the LELIE system have been found useful by most technical writers that tested the system.

In this table, we can see that the main resources which are domain dependent in LELIE are: verbs (and deverbals), fuzzy terms (of various categories) and business terms (e.g. for sentence complexity measures).

3. The development and test corpora

The modeling of the lexical knowledge acquisition proper to a domain has been realized from a number of corpora from various technical domains, authoring guidelines and genres (procedures, specifications, business rules, requirements, etc.). In a first stage, our corpora are in English, however, a similar task is planned for French. Most of the observations made for English will probably remain identical for other European languages. However, some analysis elements may differ due to the specific syntactic and morphological features.

The corpora we considered are the following:

1. Aeronautics: aircraft mechanics, airport traffic regulations, aircraft maintenance (about 110 pages, about 40 000 words);
2. Medicine: medicine test regulations (about 30 pages, about 10 000 words), bio-pharmaceutical industry (40 pages, about 11 000 words);
3. Computer science: networks and telecommunications services for the large public (about 30 pages, about 9 000 words);
4. Management: accounting software description, collaborative communication platforms (about 80 pages, about 35 000 words).

In total, our corpus has 290 pages with about 105 000 words. Aeronautics and computer science use relatively strict authoring guidelines, while medicine, pharmacy and management have more shallow guidelines and a large diversity of word uses are observed. These two areas are of much interest for the discourse and style analysis reported in section 5.

4. Acquisition of fuzzy terms

A first experimentation of the principles developed above is devoted to the case of fuzzy lexical items which is a major type of error, very representative of the need of lexical knowledge acquisition. Roughly, a lexical item is fuzzy if it denotes a concept whose meaning, interpretation, or boundaries can vary considerably according to contexts, readers or conditions, instead of being fixed once and for all. Fuzzy lexical items include several categories of adverbs (manner, temporal, location, and modal adverbs), adjectives (*adapted, appropriate*) determiners (*some, a few*), prepositions (*near, around*), a few verbs (*minimize, increase*) and nouns. These categories are not homogeneous in terms of fuzziness, e.g. fuzzy determiners and fuzzy prepositions are always fuzzy whereas e.g. fuzzy adverbs may be fuzzy only in certain contexts. The degree of fuzziness is also quite different from one term to another in a category.

The context in which a fuzzy lexical item is uttered may have an influence on its severity level. For example 'progressively' used in a short action (*progressively close the water pipe*) or used in an action that has a substantial length (*progressively heat the probe till 300 degrees Celsius are reached*) has two different severity levels because the application of 'progressively' may be more difficult to realize in the second case. In the case of this adverb, it is not the manner but the underlying temporal dimension that is

fuzzy. Finally, some usages of fuzzy lexical items are allowed. This is the case of business terms that contain fuzzy lexical items which should not trigger any error. For example, *low visibility landing procedure* in aeronautics corresponds to a precise notion, therefore 'low' must not trigger an alert in this case. The equivalent, non-business expression *landing procedure with low visibility* should probably originate an alert on 'low', but there is no consensus among technical writers. A business term is a kind of black box, but the difficulty, as shall be seen in section 5, is to precisely define what a business terms is.

A fuzzy lexical item must be contrasted with an underspecified term or expression. For example, a verb such as *damaged* in *the mother card risks to be damaged* is not fuzzy but underspecified because the importance and the nature of the damage is unknown; similarly for *heat the probe to reach 500 degrees* because the means to heat the probe are not given but are in fact required to realize the action.

In LELIE, a relatively large resource of lexical items which can be fuzzy in technical documents has been defined with an a priori severity level from 1 to 3. This resource has been elaborated from various corpora and company resources. An useful measure is the average number of fuzzy lexical items per category in an application:

Category	Total number of lexical items in lexicon
manner adverbs	210
temporal and location adverbs	167
determiners	24
prepositions	38
verbs and modals	252
adjectives	197

Table 2. Main fuzzy lexical classes: main distributions.

Since technical texts are a restricted textual genre in terms of lexical variation, the number of fuzzy lexical items per category remains manageable. The severity level depends on the context, however, some evaluation can be given a priori from experience (1 is low, 3 is high):

Category	A priori severity level
manner adverbs	2 to 3
temporal and location adverbs	in general 2
determiners	3
prepositions	2 to 3
verbs and modals	1 to 2
adjectives	in general 1

Table 3. severity level.

Finally, among these fuzzy lexical items, only a small number appear in a given document, but with rather high frequencies and in a number of diverse lexical and syntactic combinations:

Category	Average nb of terms per application
manner adverbs	18
temporal and location adverbs	11
determiners	8
prepositions	9
verbs and modals	18
adjectives	21

Table 4. frequencies in specific documents.

Table 4 shows that a domain includes on average less than 100 fuzzy lexical items from our predefined lexicon, among which about 80 items from open classes. This is a relatively small number of terms. The challenge is (1) the acquisition of the fuzzy lexical items not present in this lexicon, which represent about 40% of these items and (2) the identification of those contexts where such an item is really fuzzy (Saint-Dizier 2015).

Given these resources, our lexical acquisition system scans a given domain corpus. This corpus is composed of about 200 to 400 pages of technical documents produced in the domain considered for the lexical acquisition, this domain may be restricted to a company or a group of technical writers. The system proceeds by lexical category, as described in the following subsections.

4.1. Fuzzy determiners and prepositions

The auto-adaptative system we have defined proceeds as follows concerning closed classes (determiners, prepositions). The system first constructs the set of the items found in the domain corpora and identified as fuzzy lexical items in our fuzzy lexical items resource. These items a priori keep their fuzzy character in the application lexicon, a confirmation of their fuzzy character can be asked to e.g. the application administrator (a senior technical writer in general) together with a confirmation of their severity level. The severity level can be used to filter out items which are moderately fuzzy in the domain.

Fuzzy determiners include simple terms and compounds e.g.:*most, a majority of, almost all,* etc. Fuzzy prepositions and preposition compounds include e.g. *near, above, about, below, rather close to*, etc.

4.2. Fuzzy adverbs

The system then identifies in the domain corpora adverbs, verbs and adjectives, on the basis of a POS tagger. These three categories are identified simultaneously because they are used in syntactic analysis: adverbs are verb or adjective modifiers. Adverbs judged to be fuzzy in our database are in general fuzzy in most contexts and domains. The only variation is the severity level of each context, as illustrated at the beginning of this section, this is developed in (Saint-Dizier, 2015).

Our database contains various types of adjectives, among which:
- temporal adverbs, e.g. *momentarily, shortly,*
- manner adverbs, e.g. *rapidly, progressively,*
- quantity adverbs, e.g. *sufficiently, enough* .

Let us consider the case of adverbs. Most adverbs end by *-ly* in English. A list of adverbs can then be constructed from the domain corpus being investigated. The adverbs already present in the database of fuzzy lexical items are kept and tagged as fuzzy, and for those which are not in the database (about 55%), the system has to determine if they may have fuzzy uses. Potential fuzzy uses are identified via grammatical induction. The system searches if these adverbs are used in similar contexts than one or more elements of the database in both the domain and our development corpora to guarantee a certain coverage. By context, we have explored at the moment two situations: Adverb + Verb and Adverb + Adjective. Beyond a threshold of 70%, of identical contexts, the adverb is probably fuzzy (e.g. *carefully* exists in our lexicon while *slowly* and *cautiously* do not and are found with approximately the same usages). This threshold has been defined and tuned gradually from corpus observations. It may slightly differ from one domain to another, in particular it may depend on the number of verbs and adjectives used in a domain. These adverbs are inserted into the domain lexicon and tagged as fuzzy a priori or via a confirmation from a technical writer. At this stage the initial fuzzy lexical items database is stable and does not include this new terms. However, these new terms are kept in a 'secondary'

database for further analysis in other domains.

In a second stage, this lexicon of domain fuzzy terms can be enhanced. If these adverbs have an equivalent adjective (e.g. *careful*) or noun (*care, caution*), then these terms are also integrated into the fuzzy lexicon. Finally, recurrent expressions, recognized by a simple pattern, e.g.:

[prep, adj(fuzzy), noun],

[prep, intensifier, noun(fuzzy)],

containing these fuzzy lexical items are also considered as fuzzy expressions, e.g. *with careful precaution, with great caution*. These are included into the fuzzy term lexicon as additional fuzzy expressions. At the moment, 14 such patterns have been implemented.

This approach is relatively simple, but sufficiently accurate given the language 'profile' of technical texts. An evaluation on a 36 pages test corpus (aeronautics), was carried out, where LELIE tags all expressions it found fuzzy, with their severity level. The result was compared with the same texts manually annotated by three trained technical writers. Manually annotating fuzzy lexical items is a difficult task because annotators often have difficulties to identify all these items unless they read the text several times.

A Cohen Kappa test indicates an agreement level of 83% between the three annotators, which belong to the same company: this shows that the task is not so straightforward. In particular, some terms are judged to have a low level of fuzziness may not be tagged. Then, after discussion, a single annotated document was produced with 197 errors found. Our indicative evaluation is based on this resulting document. LELIE tagged 202 fuzzy terms, leading to a precision of 94%, and a recall of 91%. This performance level is acceptable for technical writers which do not want to be bothered too often by inappropriate error messages. A rate of less than 8 to 10% for inappropriate errors is acceptable for them, considering that their level of disagreement on such a task is around 10 to 12%, e.g. (Schriver 1989).

4.3. Fuzzy adjectives

Adjectives are not very frequent in technical documents, an analysis carried out on our corpora indicates a total of 280 adjectives, out of which about 70 are used more than 10 times. This does not include adjectives found in business terms which must not be considered as fuzzy since business terms are considered as closed terms. The main categories of these frequently used adjectives are:

- evaluative or judgment adjectives (e.g. *adequate, common, flexible, standard, useful, optimal, typical*) which are used for example to qualify a process or an equipment, adjectives such as *useless, irrelevant* are not judged to be fuzzy since they express a boolean fact,
- scalar adjectives in particular related to measures (e.g. *hot, cold, long, short, heavy*, etc.) with their corresponding antonyms,
- temporal adjectives (e.g. *recurrent, frequent*), and
- modal adjectives, e.g. *crucial, essential, necessary*.

The other types of adjectives found may be vague such as color adjectives but they are not judged to be really fuzzy. These fuzzy adjectives may be modified by an intensifier (*very, somewhat*) which reinforces their fuzzy character. Finally, adjectives such as *damaged* are not fuzzy in our approach but underspecified since the type of damage is not explicit.

When considering adjective acquisition (identified via a POS tagger) and analysis from new texts, it turns out that about 78% of these adjectives are already present in our database. The 22% are often very specialized adjectives, whose interpretation is domain dependent. These adjectives could be considered as one-word business terms. To detect whether these adjectives have a fuzzy character, a simple method consists in looking for patterns where these adjectives are combined with specific terms that reveal their fuzzy character. The main pattern is :

[adverb(+evaluative) adjective].

where evaluative adverbs include: *almost, very, rather, extremely, highly*, etc. which are typical tests for scalar adjectives.

To evaluate this analysis, let us consider a 130 page corpus from the financial software domain. The following results are obtained with our approach:

Category	Average nb of items
total nb of adjectives found	41
Fuzzy adjectives from database	23
New fuzzy adjectives	11
Not fuzzy	7

Table 5. adjective distribution.

The number of new fuzzy adjectives is relatively significant (about 1/3 of the total). In terms of human analysis, 8 among these 11 terms have been judged to be really fuzzy. The 3 others are moderately fuzzy and the expressions in which they occur can be understood. The 7 adjectives judged not to be fuzzy are indeed not fuzzy.

4.4. Fuzzy verbs and deverbals

Verbs and deverbals form a single class in technical documentation: when the agent is committed, deverbals are often used instead of passive forms which are not allowed in most CNL guidelines. For example, given:

the tester must define five use cases,

to omit the agent (assumed to be implicit), a passive form such as:

five used cases must be defined,

but since passives are not recommended in general, a deverbal can be used:

definition of five use cases.

even if the modal is not realized, the injunctive character of the statement is still present.

Our lexical database contains verbs which are anchored to their various deverbal forms. These are treated in a similar way in what concerns their fuzzy character.

Constrained natural language principles recommend to use a restricted set of verbs in a domain. Areas such as transportation and aeronautics have defined lists of recommended verbs and their uses. The set of allowed verbs is lower then 100 in most cases. However, we found areas which do not follow these recommendations where the number of verbs can go up to 600 or even 700 different verbs, not including morphological variations. We then observe many redundancies and fuzzy or vague uses. However, even with a limited number of verbs, fuzzy uses are observed, which are due to the arguments the verb is combined with. For example, *clean the screen* is judged fuzzy because it is not clear which windows on the screen must be closed. These cases are very difficult to detect without domain knowledge and inference patterns.

Let us now concentrate on the detection of fuzzy verbs or deverbals alone. In technical documents, verbs which are fuzzy are mainly:

- general purpose epistemic verbs, e.g.: *analyze, evaluate, take into account, measure, model, adapt*,
- general behavior verbs, e.g.: *avoid, allow, adopt, inspect,*

care, permit,
- general purpose action verbs, e.g.: *accelerate, enlarge, maximize,* ,
- general purpose communication verbs, e.g.: *comment, argument, document, revise*.
Although these verbs are not totally excluded from technical texts, authors are invited to use more precise verbs whenever possible. In our approach the verbs of these classes are a priori fuzzy: the technical writer is invited in the texts he produces to avoid them. The restricted set of verbs which are recommended and follow CNL principles are a priori not fuzzy. However, they may become fuzzy in combination with some arguments as illustrated above. To detect these fuzzy uses, we developed a learning mechanism that observes technical authors when they make corrections and induces fuzzy uses (Saint-Dizier 2015).
Finally, for the verbs found in a domain which are neither general purpose nor the recommended ones (when such a set exists), our approach is twofold. First they are tagged as not belonging to the recommended set when it is defined, but this does not necessarily means that these verbs must absolutely be avoided. Next, these verbs are tagged as fuzzy when they are used in several distinct contexts in the domain corpus. Context is characterized by verb arguments, in particular objects, which can be standard language or business terms. The difficulty is to evaluate that two contexts are distinct, not simply closely related terms. For that purpose, a domain terminology (or ontology) is necessary, on which a distance metrics is applied. Then, a verb is fuzzy if one of these conditions is met:
- the objects of the verb are general purpose terms in the ontology (characterized by a distance of N arcs to reach the leaf node, to be adjusted depending on the domain ontology complexity, which may have a hierarchy of up to 9 nodes),
- the objects of the verb may be low level words in the terminology, but they belong to at least 3 different subtrees, where the common ancestor is at least 3 nodes up in the hierarchy,
- the objects of the verb include more than a threshold T of different business terms. T depends on the complexity and the number of the business terms. A value around 10 seems to be a good indicator.
Similarly to adverbs and adjectives, we asked a group of three technical writers to annotate the verbs they think are fuzzy in their context in a 36 page long document, including general purpose verbs. 76 occurrences of fuzzy verbs were found in total with a kappa of 83% which is relatively low. Our system detected 82 fuzzy verbs, the resulting accuracy is 85%. Given the kappa, this result sounds satisfactory. Indeed, even if our system erroneously tags verbs as fuzzy, this may be useful to help technical writers to improve their writing skills and concentrate as mush as possible on the use of verbs which are unambiguous, not polysemous and probably with a low degree of fuzzyness.

5. Acquisition of Business Terms

Automatically acquiring business terms is a major challenge. This is an important task for several reasons in LELIE, e.g. (1) fuzzy terms in business terms must not generate any error message, (2) business terms count for a single unit when the length of a sentence is evaluated, (3) they also enter with a specific weight when the complexity of a sentence is measured. Besides their use in LELIE, it is often crucial for a company or a particular activity to have the list of the business terms which are used, to structure it and to possibly eliminate redundancies or useless uses.
The first difficulty is to define what a business term is and what is its structure. For example:
flap retraction speeds
is probably a business term in aeronautics, but are lexical variations such as:
minimum flap retraction speeds,
maximum flap 5' retraction speed or
25 degrees flap retraction speed
business terms ? If so, this means that quite a large level of lexical variation and syntactic composition may occur within business terms. Finally, is a single word with a specific meaning and uses a business term or just a polysemous word that needs contextual interpretation ?
On the same corpus as for fuzzy lexical items, we carried out an additional experiment, with the same annotators, asking them to identify business terms. On 36 pages of text, 1152 business term occurrences have been found, with 291 different terms, union of all the occurrences found by the annotators. The kappa test indicates a level of agreement of only 81% among the three annotators. Again, this shows that this task is difficult.
Roughly, the auto-adaptative system uses the following criteria to identify business terms:

1. Detect structures from 3 to 6 terms not containing any closed class term (modals, auxiliaries, determiners, conjunctions, prepositions) and no inflected verb: these are potential candidates for business terms. 6 terms seems to be the largest size for a business term. All terms, even with some overlap, are kept.

2. Search in the corpora for variations, as illustrated above, and evaluate the degree of lexical variation at any position in the business term considered. Lexical variation is characterized by two types of patterns:

   - general purpose patterns, such as the inclusion of an adjective in the pattern: *flap retraction speeds* $\rightarrow$ *flap retraction maximum speeds*. These patterns are implemented in Dislog, which runs on our TextCoop development environment (free resource). Five general purpose patterns have been defined, a single one manages the different positions an adjective may have in a business term.
   - specific patterns introducing e.g. numbers, units, adjectives or additional nouns at specific places. These patterns are induced and generalized from the observation of already existing sets of business terms and their variations. For example: *flap retraction speeds* $\rightarrow$ *5 degree flap retraction speed*. These patterns are much more ad'hoc and need to be developed with care. So far, we have developed 27 such pattern, with relatively high lexical category, e.g.:
     `[noun]` $\rightarrow$ `[number unit noun].`

3. evaluate the level of variation between pairs of terms. If it is too high (according to our evaluation, which still needs to be adjusted: above 6 different forms for a 3 word term, above 8 forms for a 4 word term), then it is probably not a business term, to be confirmed by the administrator. This is a kind of measure of the variability level of an expression.

4. Search in the corpora for expanded versions or these structures, i.e. if a business term is composed of the words A B C D, search via patterns for forms where these words occur in a different order with grammatical realizations, in particular using prepositions and determiners, e.g. for the above example: *the minimum speeds for the retraction of flaps*. Such an extended reformulation is indeed not allowed in the case of business terms, therefore such terms are eliminated.

5. the terms selected in (1) above that pass the various tests described in (2), (3) and (4) above are a priori business terms in the domain considered.

Considering the low level of agreement between annotators, it is difficult to produce a very accurate evaluation of the results we obtained. To have a more independent evaluation, the domain we considered (energy) does not have any approved terminology. The results can be summarized as follows:

- number of different business terms recognized by annotators (A): 291,
- number of different business terms recognized by LELIE (B) : 315,
- number of similar business terms (C): 269,
- number of different business terms: 76,
- agreement level computed via the formula: C / ((A+B)/2: 88%.

The result obtained by the auto-adaptative system is relatively good considering the complexity of the task. However, a larger evaluation is ongoing with a larger set of potential business terms, on two domains (aeronautics and bio-pharmacy). New decision criteria than the three ones developed above may probably emerge.

6. Induction of the most common Discourse Structures

This task has quite a different purpose in the auto-adaptative landscape. It aims at automatically identifying the probably implicit authoring behaviors of technical writers, and to induce a set of best practices in terms of discourse organization. This is an important feature of our LELIO system. LELIE and TextCoop (the discourse processing platform on which LELIE runs) can recognize a variety of general purpose discourse structures often found in technical texts: conditions, circumstances, justifications, illustrations, reformulations, elaborations, purposes, etc. and structures specific to technical texts: requirements, instructions and titles, prerequisites, warnings, advice, etc.

The induction of the most frequent discourse structures is carried out by making statistics on the frequency and the position of these structures in sentences (mainly instructions and requirements for the general purpose structures) and paragraphs, and indicates the preferred positions. The most frequent positions are then proposed per structure type as guidelines or as by-default positions. The goal is to make texts more homogeneous, with a higher cohesion, so that they are easier to understand by readers. Results largely depend on the application, however, some general practices are observed.

7. Perspectives

The research reported in this paper has a direct application in LELIE and in many other authoring tools to acquire large portions of the lexical knowledge that is needed to customize a kernel such as LELIE to an application domain. This task is still often done manually by lexicographers, however, this task is long, costly and error prone. The results given in this paper, probably for the most crucial and complex lexical resources (business terms and fuzzy lexical items) show that it is possible automate this process. Obviously, a human validation will always be necessary, but the induced workload is much lower.

Another element, briefly developed in this paper, is the automatic induction of the main discourse structures used in a domain or application or group of technical writers. The goal is to contribute to improving style homogeneity and cohesion. This is a major feature to facilitate the reading of technical texts, leaving the cognitive load for more central tasks.

These experiments remain quite empirical and probably need to be adjusted to different context, including the language and conceptual complexity of the domain and the critical level of the area in terms of risks for example. These experiments constitute the first step of the development of the LELIO platform, a generator of LELIE customized solution, where LELIE is a kernel that needs to be adapted on several levels to be really useful in an industrial context.

8. Acknowledgments

I thank Juyeon Kang for useful discussions a year ago that contributed to motivating this research. I am grateful to the CNRS, my institution, for letting me pursue this work. I also thanks the companies that contributed to the project via documents or analysis from technical writers. I however regret the large difficulties raised by our private valorisation institution (toulouse tech transfer) and an SME who made this project really difficult in terms of juridical and administrative constraints.

9. Bibliographical References

Arora, C., Sabetzadeh, M., Briand, L., Zimmer, F., and Gnaga, R. (2013). Automatic Checking of Conformance to Requirement Boilerplates via Text Chunking: An Industrial Case Study, 7th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM 2013). Baltimore, MD, USA.

Barcellini F, Albert, C., and Saint-Dizier, P. (2012). Risk Analysis and Prevention: LELIE, a Tool dedicated to Procedure and Requirement Authoring, LREC, Istanbul.

Fuchs, N.E. (2012). First-Order Reasoning for Attempto Controlled English, In Proceedings of the Second International Workshop on Controlled Natural Language (CNL 2010), Springer.

Ganier, F., and Barcenilla J. (2007). Considering users and the way they use procedural texts : some prerequisites for the design of appropriate documents. In D. Alamargot, P. Terrier and J.-M. Cellier (Eds), Improving the production and understanding of written documents in the workplace, Elsevier Publishers.

Garnier, M. (2011). Correcting errors in N+N structures in the production of French users of English, EuroCall, Nottingham.

Grady, J. O. (2006). System Requirements Analysis, Academic Press, USA.

Hull,E., Jackson,K., and Dick, J. (2011). Requirements Engineering, Springer.

Kuhn, T. (2014). A Survey and Classification of Controlled Natural Languages. Computational Linguistics, 40(1).

Saint-Dizier P. (2012). DISLOG: A logic-based language for processing discourse structures, in proceedings of LREC 2012.

Saint-Dizier P. (2014). Challenges of Discourse Processing: the case of technical documents. Cambridge Scholars, UK.

Saint-Dizier P. (2015). Features of an Error Correction Memory to Enhance Technical Texts Authoring in LELIE, IJKCDT journal, 5(2).

Scherrer, Y. (2007), Adaptive String Distance Measures for Bilingual Dialect Lexicon Induction, ACL 2007.

Schriver, K. A. (1989). Evaluating text quality : The continuum from text-focused to reader-focused methods, IEEE Transactions on Professional Communication, 32, 238-255.

Van der Linden K (1993). Speaking of Actions: choosing Rhetorical Status and Grammatical Form in Instructional Text Generation. PhD, Univ. of Colorado, USA.

Weiss E. H. (2000). Writing remedies. Practical exercises for technical writing. Oryx Press.

White, C., Schwitter, R. (2009). An Update on PENG Light. In: Pizzato, L., Schwitter, R. (eds.) Proceedings of ALTA 2009, Sydney, Australia, pp. 80-88.

Wyner, A., et ali. (2010). On Controlled Natural Languages: Properties and Prospects.